

Video Game Music Generator

...

Robert Tung, Aaron Plourde, Katherine Tan

Overview

- Read in XML and parse for MIDI note values and durations
- Determine overall structure and motifs of piece
- Generate new motifs for each motif in melody and generate rest of piece
 - Notes and durations
- Generate harmony and drums
- UI input and audio output!

Reading in and parsing XML

- Look for tags:
 - `<measure, args...> </measure>`
 - `<note> ... </note>`
 - `<pitch> ... </pitch>`
 - `<step> ... </step>`
 - `<octave> ... </octave>`
 - `<rest />`
 - `<alter> ... </alter>`
 - `<instrument id, ... />`
 - `<duration> ... </duration>`
 - `<measure, args...> </measure>`
- Add values, grouped by measure, to an array.
 - each index contains `[voice#, midi_note#, note_length]`

Determining motifs of piece

- Find repeated sets of measures in a piece
 - Go through non repeat sections to find set of all repeats
 - Does this by taking in an array of measure arrays of notes and manipulating an array of flags for each measure
 - Returns a flag array where each flag corresponds to a motif number
 - Example:
 - Input: `[[1, 1, 1], [2, 2, 2], [3, 3, 3], [1, 1, 1], [2, 2, 2], [4, 4, 4]]`
 - `[1, 1, 1], [2, 2, 2]` is a repeated pattern
 - Output flag array: `[1, 1, 2, 1, 1, 3]`
 - Output motif dictionary: `[(1 -> [0, 2]), (2 -> [2, 3]), (3 -> [5, 6])]`

Generate Melody

- For each isolated motif:
 - Durations generated through Markhov models to fill the bars
 - Notes generated by Markhov models afterward
 - New dictionary generated for each motif
- Track repeated motifs to not re-generate randomization each time it is found
- Follow some motifs with scrambled versions of motifs

Generate Drums

- Finds the most similar drumline generated from a cellular automata rule to the given drumline
 - If none given, randomly chooses a cellular automata rule

Generate Harmony

- Pulls the note that was above the melody note when the melody note played originally
 - If none are available, it just uses the note an octave below.

Future extensions

- Generate music that lines up multiple voices
 - Code already parses all voices, generation code just can't use them all yet
 - This will also increase harmony complexity
- Parse out drums from XML into same format as used to create the “most similar” cellular automata rule
 - Code to compare similarity already exists, just need to be able to parse drums into right format
- Output as XML/MIDI
- Vary SynthDefs

Code Demo

...